VT-MCNet: High-Accuracy Automatic Modulation Classification Model based on Vision Transformer

Thien-Thanh Dao, Dae-Il Noh, Quoc-Viet Pham, *Senior Member, IEEE*,
Mikio Hasegawa, Hiroo Sekiya, and Won-Joo Hwang, *Senior Member, IEEE*

***Abstract*—Cognitive radio networks' evolution hinges significantly on the use of automatic modulation classification (AMC). However, existing research reveals limitations in attaining high AMC accuracy due to ineffective feature extraction from signals. To counter this, we propose a vision-centric approach employing diverse kernel sizes to augment signal extraction. In addition, we refine the transformer architecture by incorporating a dual-branch multi-layer perceptron network, enabling diverse pattern learning and enhancing the model's running speed. Specifically, our architecture allows the system to focus on relevant portions of the input sequence, thus, it improves classification accuracy for both high and low signal-to-noise regimes. By utilizing the widely recognized DeepSig dataset, our pioneering deep model, termed as VT-MCNet, outshines prior leading-edge deep networks in terms of classification accuracy and computational costs. Notably, VT-MCNet reaches an exceptional cumulative classification rate of up to 99.24%, while the state-of-the-art method, even with higher computational complexity, can only achieve 99.06%.**

***Index Terms*—Modulation classification, convolutional neural network, wireless communications, vision transformers.**

I. Introduction

In wireless communication, optimizing transmission rates often require adaptive modulation schemes in changing channels. This necessitates exchanging modulation information between transmitter and receiver, adding protocol overhead. To reduce this, it's beneficial if the receiver can autonomously determine modulation types without prior knowledge. Automatic Modulation Classification (AMC) was introduced for this purpose, acting as an intermediary step between signal detection and demodulation. AMC identifies modulation formats, even in noisy or interfered signals, providing robust recognition.

In recent years, deep learning (DL) has emerged as a potent tool for AMC, with researchers developing various neural networks to enhance performance. The RadioML2018.01A dataset for AMC, along with custom network architectures like VGG and ResNet, was introduced in [1]. The performance of radio classification varied when assorted convolutional neural network (CNN) architectures were scrutinized, as the CNN processes data through multiple convolutional layers, it learns hierarchical representations of the In-phase/Quadrature (I/Q) signals. Another novel data-driven AMC method based on Long Short-Term Memory (LSTM) is presented in [2], [3]. Moreover, a composite CNN-LSTM-based method is advocated in [4]. In [5], [6], novel CNN architectures for robust AMC are proposed. [7] presented a practical threat framework and devised an innovative intra-class universal adversarial perturbation (IC-UAP) method, aimed at challenging deep learning-based modulation classifiers. These studies showed that CNNs can extract meaningful discrimination features from multiscale representations for AMC tasks.

Transformers [8], built upon the principles of self-attention, have recently become a standard in natural language processing, and vision transformers [9] have been extensively employed for image classification tasks. However, their application in signal classification remains constrained. The closest works to our study are [10] and [11]. While [10] demonstrates the utility of transformer blocks, it still exhibits limited accuracy in low signal-to-noise (SNR) conditions. Conversely, [11] transforms input I/Q signals into a matrix format, converting patches into sequences for the transformer architecture, resulting in improved classification accuracy at low SNR scenarios. However, both these studies compute feature embedding through a linear operation on the input signal and employ full-attention blocks, potentially losing continuous signal details like phase, amplitude, and frequency variations. In contrast, our research advances beyond previous work by incorporating ViT transformer blocks with multiple kernel sizes. This approach allows our proposed model to extract a more diverse set of features compared to a single kernel, enhancing the accuracy of AMC.

The main contributions of this study are summarized below:

- We provide a deep analysis of high-accuracy AMC and prove that different kernel sizes are more effective at capturing feature information from the input signals.
- We develop an optimized high-accuracy AMC model, namely ViT-based AMC Network (VT-MCNet), achieved by analyzing the representation ability of each transformer layer, striking a better trade-off between speed and classification accuracy.
- We incorporate robustness into transformer blockheads to improve signal classification performance, cumulative accuracy of 99.24%, and maintain a fast running speed.

T.-T. Dao is with the Department of Information Convergence Engineering, Pusan National University, Busan 46241, South Korea (e-mail: daothanh@pusan.ac.kr). D.-I. Noh is with the Department of Information Convergence Engineering, Center for Artificial Intelligence Research, Pusan National University, Busan 46241, South Korea (e-mail: nohdi1991@pusan.ac.kr). Q.-V. Pham is with the School of Computer Science and Statistics, Trinity College Dublin, Dublin 2, D02 PN40, Ireland (e-mail: viet.pham@tcd.ie). M. Hasegawa is with the Department of Electrical Engineering, Tokyo University of Science, Tokyo 162-8601, Japan (e-mail: hasegawa@haselab.ee.kagu.tus.ac.jp). H. Sekiya is with the Graduate School of Engineering, Chiba University, Chiba 263-8522, Japan (e-mail: sekiya@faculty.chiba-u.jp). W.-J. Hwang (corresponding author) is with the Department of Information Convergence Engineering, Center for Artificial Intelligence Research, Pusan National University, Busan 46241, South Korea (e-mail: wjhwang@pusan.ac.kr).

Following are results of a study on the "Leaders in INdustry-university Cooperation 3.0" Project, supported by the Ministry of Education and National Research Foundation of Korea

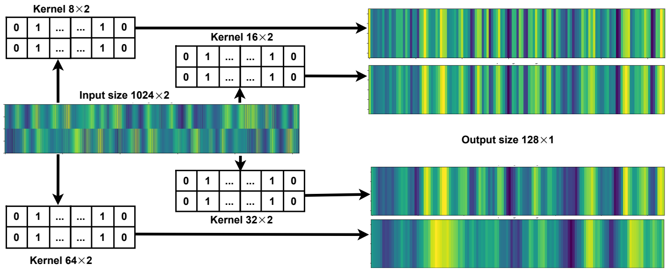

Fig. 1. Visualization of multi-kernel sizes.

II. Methodology

Classification accuracy and running speed are important in AMC, and finding a balance between these factors is critical. In this section, we explore signal representations of convolution operations and a novel transformer-based signal classification approach to effectively overcome this challenge.

A. Analysis of High Accuracy Performance

To analyze the effect of kernel size, Fig. 1 shows the output of four convolutional layers with fixed kernel sizes from 8×2 to 64×2 applied to a 1024×2 I/Q signal. The output indicates that the kernel size plays a crucial role in feature extraction. With a small kernel size 8×2, the convolution mainly interacts with neighboring pixels, offering a detailed, localized view. Thus, it is sensitive to short-term temporal patterns or high-frequency components. Although adept at delineating intricate details, this kernel size can be ensnared by noise, causing potential misconstrual, especially amidst noise-rich datasets. A larger kernel size such as 16×2, 32×2, and 64×2 can capture broader patterns, providing a more panoramic perspective of the input data. Notably, these large kernels are especially proficient in capturing longer-term temporal dependencies, low-frequency components, and amplitude variations. Nevertheless, applying large convolutions to CNNs can result in performance and speed degradation. As a result, it becomes essential to utilize multiple kernel sizes to leverage the advantages of both small and large kernel sizes.

B. VT-MCNet Design

The ViT [9] exhibits scalable architectures and proficiency in capturing global features. However, extraction conventional vision transformer-based methods commonly employ convolution 3×3 projection during the feature embedding phase for image classification. Adapting this approach to signal classification can result in overlooking key signal characteristics as explained in Section II-A. To offset these limitations and boost global feature extraction, we introduce the notion of a multi-kernel block and engineer it to encapsulate more signal features within a single token.

Multi-kernel block (MTK): The signal input, denoted as a pair of complex vectors $r \in \mathbb{C}^{L \times 1}$ and referred to as I/Q components, is first concatenated to create a data matrix $R \in \mathbb{C}^{L \times 2}$. Then, R is reshaped into the tensor I with dimensions $I \in \mathbb{C}^{1 \times L \times 2}$, where L symbolizes the length of the signal and is set to 1024 in this work. The multi-kernel block is then engaged, featuring four parallel processing units each with a predefined kernel size of $[8 \times 2]$, $[16 \times 2]$, $[32 \times 2]$, $[64 \times 2]$. These sizes allow for the capture of information at varying levels of detail, as depicted in Fig. 2. Each unit performs a convolution operation with a kernel $K \in \mathbb{R}^{C_{out} \times K_w \times K_h}$, where K_w, K_h are the kernel width and height, respectively, and C_{out} is the output channels, alongside the specified padding P and stride S. The resulting output tensor can be characterized as follows:

$$\begin{aligned}
\mathrm{O}_0 &= \mathrm{Conv}\left(I, W_0, b_0; K_0 = (C_{out}, 8, 2), \mathrm{S}, \mathrm{P}_0 = (0,0)\right), \\
\mathrm{O}_1 &= \mathrm{Conv}\left(I, W_1, b_1; K_1 = (C_{out}, 16, 2), \mathrm{S}, \mathrm{P}_1 = (4,0)\right), \\
\mathrm{O}_2 &= \mathrm{Conv}\left(I, W_2, b_2; K_2 = (C_{out}, 32, 2), \mathrm{S}, \mathrm{P}_2 = (12,0)\right), \\
\mathrm{O}_3 &= \mathrm{Conv}\left(I, W_3, b_3; K_3 = (C_{out}, 64, 2), \mathrm{S}, \mathrm{P}_3 = (32,0)\right),
\end{aligned}$$

where W_i and b_i, $i = \{0, 1, 2, 3\}$, are the weights and bias terms of the convolutional operation, respectively. A stride of $\mathrm{S} = (8, 2)$ is applied to all convolutions. Finally, the output of the multi-kernel block is obtained by concatenating, denoted as Concat, the output tensors from all convolutional layers along the channel dimension as $X_0 = \mathrm{Concat}\left(\mathrm{O}_0, \mathrm{O}_1, \mathrm{O}_2, \mathrm{O}_3\right)$.

Position Embedding: Following the multi-kernel block operation, we obtain an output layer defined as $X_0 \in \mathbb{R}^{C_{out} \times H_{out} \times 1}$, where H_{out} signifies the output height and is computed as $H_{out} = \frac{L-K+2P}{S} + 1$, while C_{out} represents the total output channels. Subsequent to this, a linear position embedding is created by flattening the transposed output X_O^T. The next step involves concatenation with the class token $x_{cls} \in \mathbb{R}^{1 \times C_{out}}$ and coupling with the learnable positional bias $E_{pos} \in \mathbb{R}^{(H_{out}+1) \times C_{out}}$. This entire procedure can be symbolically represented as $X_1 = \mathrm{Concat}\left(x_{\mathrm{cls}}, X_O^T\right) + E_{pos}$.

Transformer Encoder: The transformer encoder is composed of alternating layers of multi-headed self-attention (MHSA) and multilayer perception (MLP) blocks. Layer normalization (LN) is implemented before each block, and residual connections are established after each block.

Multi-Head Self-Attention (MHSA): is a fundamental component in transformer architectures, built upon the concept of "query-key-value" (qkv) self-attention (SA). SA allows the model to consider and incorporate input features based on their relationships, regardless of their sequential order. In this study, an input sequence $X_1 \in \mathbb{R}^{(H_{out}+1) \times C_{out}}$ is utilized. This sequence enables the calculation of a weighted sum of all elements in the sequence through the construction of three vectors: the Query vector ($Q = X_1 W_Q$), the Key vector ($K = X_1 W_K$), and the Value vector ($V = X_1 W_V$). The attention weights between each pair of elements are then computed as the dot product of their respective query and key vectors, normalized by the square root of the key vector's dimension, and passed through a softmax function:

$$\mathrm{SA}(X_1) = \mathrm{softmax}\left(QK^T / \sqrt{H_{out+1}}\right) \cdot V, \tag{1}$$

where W_Q, W_K, and W_V are trainable weight matrices.

The MHSA enhances SA by using multiple parallel self-attention operations, referred to as "heads". Each head computes a unique learned linear projection of the input, resulting in diverse versions of self-attention being computed. The outputs of all the heads are then concatenated and linearly transformed to generate the final output, expressed as: $\mathrm{MHSA}(z) = \mathrm{Concat}\left(\mathrm{SA}_1(X_1); \mathrm{SA}_2(X_1); \ldots; \mathrm{SA}_{N_h}(X_1)\right)$, where N_h is the number of self-attention heads.

Dual-branch MLP (DB-MLP): To improve the transformer performance, we proposed dividing the MLP input into two

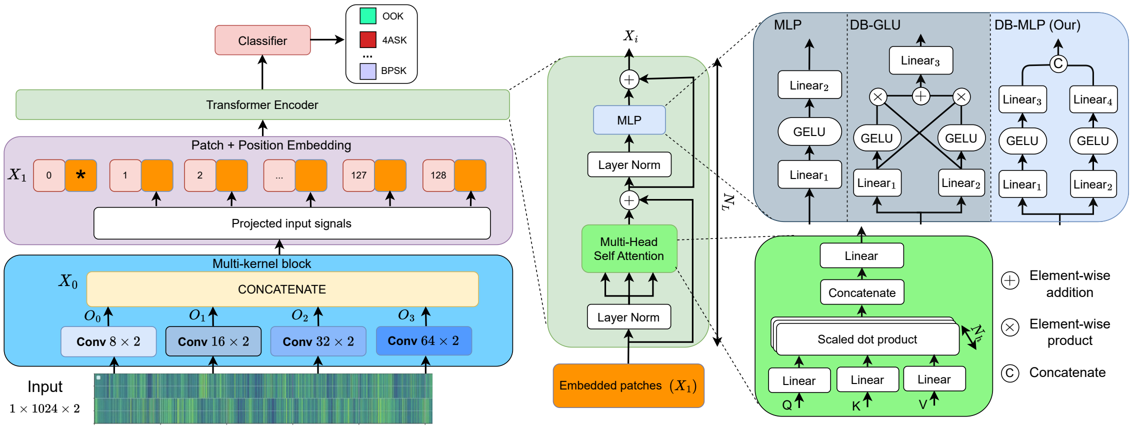

Fig. 2. Model overview. Other structures are depicted in the MLP module for comparison purposes, but only DB-MLP is utilized in our model architecture.

TABLE I
DATASET RADIOML2018.01A DESCRIPTION.

No. modulation modes	24	No. samples	2.555.904
No. SNRs/modulation	26	Sample's shape	1024×2
No. samples/SNR	98.304	No. samples/module	4096

and applying separate linear transformations, it facilitates diverse pattern learning while maintaining parameter efficiency. This dual-branch structure amplifies model capacity and flexibility without significant computational burden, courtesy of its parallel processing suitability. Its unique design enables a broader, specialized capture of data characteristics, offering a competitive edge in tackling intricate data patterns. In our proposed dual-branch enhanced MLP, we split the output tokens of MHSA z into two equal parts along the last dimension, referred to as $m, n = \text{Split}(\text{MHSA}(z))$. Linear projection is then utilized to generate DB-MLP, as detailed in the following

$$\begin{aligned} \text{DB-MLP} = \text{Concat}(&W_{fc4} \cdot \text{GELU}(W_{fc2} \cdot m), \\ &W_{fc3} \cdot \text{GELU}(W_{fc1} \cdot n)), \end{aligned} \tag{2}$$

where $W_{fc1}, ..., W_{fc4}$ are learnable parameters of linear transformations, referred to as $Linear_1, ..., Linear_4$ in Fig. 2.

III. EXPERIMENTAL, RESULTS, AND DISCUSSION

A. Dataset Description and Implementation Details

We assessed VT-MCNet using RadioML2018.01A. This dataset covers prevalent communication system impairments as detailed in Table I. The dataset was divided into training, validation, and testing sets in a 6:2:2 ratio, respectively. We employed the PyTorch library for training. The model trained for 150 epochs, starting at a learning rate of 10^{-3} and decreasing by a factor of 0.8 every 20 epochs using the Adam optimizer [12]. We employed categorical cross-entropy as the loss function and used a batch size of 1024.

B. Results

This section presents simulation results for VT-MCNet, comparing its accuracy to benchmark schemes across different SNRs, as shown in Fig. 3. VT-MCNet outperforms other SOTA network architectures, including [2], [10]. The performance gap is more pronounced at lower SNRs, where our network achieves an accuracy of approximately 63.4% at 0 dB, a figure that is substantially higher by at least 7.8% compared to other methodologies such as 53.8% in [10], 55.6% in [2]. At 24 dB, VT-MCNet displays an improvement of 0.18% over the baseline model, FEA-T [10], which previously demonstrated the best performance amongst the traditional models.

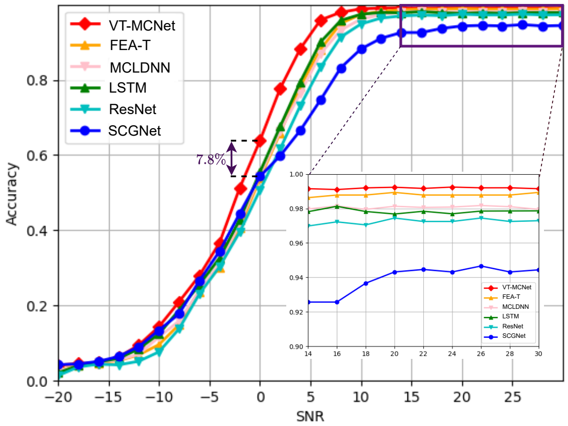

Fig. 3. Accuracy of different benchmarks under various SNRs.

Fig. 4(b) depicts a confusion matrix of 24 modulations at 0dB. Since the signal and noise powers are equal therein, confusion matrix errors occur for most modulations. The dominant noise obscures modulation features, leading to significant misclassification as signals become easily distorted. At 24dB in Fig. 4(c), VT-MCNet achieves almost perfect classification for low-level modulations like FM, ASK, MSK, and PSK, with minimal errors. Some confusion matrix errors occur for 64QAM, 128QAM, and 256QAM due to their hierarchical relationship. Errors are also observed in SSB-WC with SSB-SC modulations and DSB-WC with DSB-SC modulations. Based on these observations, the VT-MCNet model may struggle to classify analog signals, suggesting that more effective analog signal characteristics can be further considered to improve AMC performance.

We compare the computational complexity of VT-MCNet against SOTA deep learning networks, utilizing floating point operations per second (FLOPs) and trainable parameters. As illustrated in Table II, VT-MCNet manages to have the lowest FLOPs among many SOTA benchmarks. Notably, VT-MCNet shows significant improvement over [10] in terms of trainable parameters (148K vs 177K), besides its superiority in classification accuracy (64.8% vs 61.81% in average). VT-

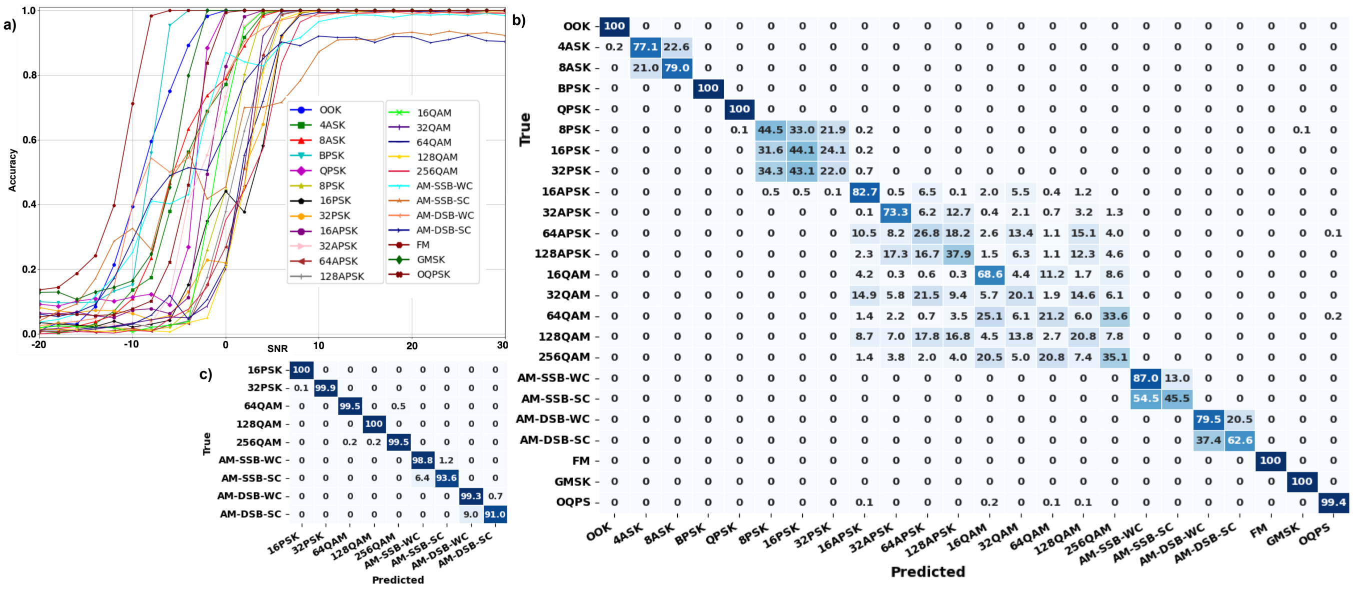

Fig. 4. Accuracy of modulation classification over different SNR levels (a), confusion matrices for 24-modulation classification at (b) 0 dB SNR (Accuracy: 63.4%) and (c) +24 dB SNR (Accuracy: 99.24%, we only display classes that have misclassifications).

MCNet requires model parameters than the SCGNet model, which only achieves an accuracy of 58.82% on average and 94.39% at most, both are far from VT-MCNet performance. Furthermore, VT-MCNet surpasses all the benchmarks in both average and maximum accuracy, achieving 64.8% and 99.24%, respectively. Consequently, VT-MCNet proves to be an efficient solution for AMC tasks, skillfully balancing computational cost and feature extraction. This balance enhances the classification accuracy of the model while maintaining computational resource consumption at an optimal level.

C. Ablation Study

We study the impact of hyper-parameters on the performance of AMC in deep networks. Specifically, we focus on key hyperparameters such as kernel dimensions in MTK, number of layers, heads within the transformer encoder, and hidden size of MLP. The objective of conducting a series of comparative experiments is to analyze the impact of variations in these hyperparameters on VT-MCNet's performance.

1. Analysis number of layers (N_L), hidden size (H_{mlp}), and number of heads (N_h): To optimize the transformer architecture in our VT-MCNet model, we need to understand the parameter distribution. Notably, while maintaining the model complexity at the same level, an increase in the number of parameters also results in a corresponding increase in the number of FLOPs. In the transformer encoder, including the MHSA and DB-MLP modules, each MHSA employs three distinct weight matrices: W_Q, W_K, and W_V that have dimensions of $C_{out} \times (C_{out}/N_h)$. Consequently, the total number of parameters per attention head is given by $3 \times C_{out} \times (C_{out}/N_h)$. Considering all attention heads, the cumulative parameters amount to $N_h \times 3 \times C_{out} \times (C_{out}/N_h)$. Additionally, the output undergoes further processing via a linear layer with a weight matrix of dimensions $C_{out} \times C_{out}$, contributing $C_{out} \times C_{out}$ parameters. Combining these elements, the total number of parameters in the self-attention mechanism is expressed as:

$$P_1 = N_L \times ((N_h \times 3 \times C_{out} \times (C_{out}/N_h)) + C_{out}^2), \quad (3)$$

TABLE II
COMPARISON COMPUTATIONAL COMPLEXITY AND ACCURACY OF DIFFERENT CNN ARCHITECTURES

Network	Size (params.)	FLOPs (M)	Avg/Max Acc (%)
SCGNet [13]	**107K**	36	58.82/94.39
OTA-Resnet [1]	236K	88	60.08/97.46
MCLDNN [14]	407K	235	61.86/98.20
LSTM [2]	203	38	62.51/98.22
FEA-T [10]	177K	55	61.81/99.06
VT-MCNet	148K	**32**	**64.8/99.24**

TABLE III
MODEL PERFORMANCE IN DIFFERENT HYPER-PARAMETER

N_L	N_h	C_{out}	H_{mlp}	Size	FLOPs	Accuracy
					(M)	0dB − 30dB/Max
4	4	64	128	115K	22.2	94.44/99.20
4	4	64	192	132K	24.3	94.63/99.24
4	4	64	256	148K	26.4	94.73/99.25
4	4	96	128	222K	39.6	94.52/99.21
6	4	64	128	166K	33.1	94.01/99.19
6	6	72	128	139K	26.2	94.66/99.22

In DB-MLP, the parameter count is calculated using equation (2), where $W_{fc1} \in \mathbb{R}^{\frac{C_{out}}{2} \times \frac{H_{mlp}}{2}}$. Since there are 4 linear projections in N_L layers, the number of parameters becomes:

$$P_2 = N_L \times (4 \times (C_{out}/2) \times (H_{mlp}/2)). \quad (4)$$

Our study aims to explore variations in the model within a specific parameter range of 130K to 200K. This constraint facilitates a comprehensive analysis of different model scales, taking into account the related computational complexity and resource restrictions. Equations (3) and (4) reveal that parameters N_L, N_h, and C_{out} have a more significant impact on model size and FLOPs than H_{mlp}. However, the results in Table III indicate that increasing N_L, N_h, and C_{out} does not substantially improve accuracy. In contrast, increasing the H_{mlp} parameter noticeably enhances model accuracy. The evidence suggests that selecting H_{mlp} = 192 rather than H_{mlp} = 256 provides nearly optimal accuracy and significantly reduces both the computational demand and resource consumption.

TABLE V
COMPARISON COMPUTATIONAL COMPLEXITY AND ACCURACY OF CONVENTIONAL MLP, DB-GLU [10] AND DB-MLP.

Network	Size (params.)	FLOPs (M)	Avg/Max Acc (%)
MLP	181K	30.5	63.5/99.2
DB-GLU [10]	148K	26.4	64.6/99.2
DB-MLP	**132K**	**24.3**	**64.8/99.24**

TABLE IV
MODEL PERFORMANCE IN DIFFERENT KERNEL SIZES.

Kernel	FLOPs	Throughput rate	Accuracy
Size (K)		(samples/sec)	0dB − 30dB
8x2	23.98	1600	91.21/98.33
16x2	24.11	1606	92.84/99.11
32x2	24.37	1625	93.63/99.16
64x2	24.9	1640	93.87/99.18
MTK	24.3	1621	**94.63/99.24**

2. Kernel size study: We analyze multiple kernel sizes to select the optimal multi-kernel size for the AMC model. We consider the input channels C_{in}, output channels C_{out}, and bias term. The number of parameters is $P = (K_{width} \times K_{heigh} \times C_{\text{in}} + 1) \times C_{\text{out}}$. Since C_{in} and C_{out} are unchanged, our attention is turned towards modulating the kernel size in an attempt to optimize our multi-kernel structure. Both single-kernel and multi-kernel sizes are evaluated, as shown in Table IV, showing a gradual increase in accuracy with an expansion in kernel size. However, it is worth noting that increasing the kernel size escalates the number of parameters. For this reason, our model applies a multi-kernel size, which not only curtails the model size but also improves feature extraction from the input signal, as elaborated in Section II-A.

3. DB-MLP study: We compared our modified transformer with SOTA architectures by assessing computational complexity and accuracy, as shown in Table V. The architectures considered in our evaluation were the conventional MLP, DB-GLU in [10], and our proposed DB-MLP. Table V indicates that the DB-MLP module achieved the highest accuracy among the three architectures. As analytically analyzed in Section II-B, the DB-MLP utilizes a linear projection with half the size of the original MLP. Consequently, the number of parameters in the DB-MLP is approximately 50% less than the MLP and 25% less than the DB-GLU scheme. Notably, our proposed DB-MLP architecture shows a superior speed in terms of FLOPs compared to MLP and DB-GLU. These findings highlight our DB-MLP as the optimal feed-forward network for the transformer in the context of the AMC task.

IV. CONCLUSION

We have presented a highly accurate CNN architecture for identifying different modulation modes in communication networks. By incorporating multi-kernel blocks, our approach can capture signal information globally and locally so as to improve classification accuracy. Additionally, we have investigated a transformer encoder to reduce the computational complexity without sacrificing accuracy. Experimental results on the RadioML2018.01A dataset have demonstrated that our deep model, VT-MCNet, achieves 99.24% recognition accuracy for 24 modulation types at high SNRs. Compared to many other DL architectures, our proposed approach delivers significant performance improvements with reasonable computational costs. In our future work, we aim to employ automated hyperparameter tuning methods like grid search or Bayesian optimization to enhance model configurations efficiently.

REFERENCES

[1] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.

[2] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, May. 2018.

[3] M. M. Elsagheer and S. M. Ramzy, "A hybrid model for automatic modulation classification based on residual neural networks and long short term memory," *Alex. Eng. J.*, vol. 67, pp. 117–128, Mar. 2023.

[4] Z. Zhang, H. Luo, C. Wang, C. Gan, and Y. Xiang, "Automatic modulation classification using CNN-LSTM based dual-stream structure," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13 521–13 531, Nov. 2020.

[5] T. Huynh-The, C.-H. Hua, Q.-V. Pham, and D.-S. Kim, "MCNet: An efficient CNN architecture for robust automatic modulation classification," *IEEE Commun. Lett.*, vol. 24, no. 4, pp. 811–815, Apr. 2020.

[6] T. Huynh-The, Q.-V. Pham, T.-V. Nguyen, T. T. Nguyen, D. B. d. Costa, and D.-S. Kim, "RanNet: Learning residual-attention structure in CNNs for automatic modulation classification," *IEEE Wireless Commun. Lett.*, vol. 11, no. 6, pp. 1243–1247, Mar. 2022.

[7] R. Li, H. Liao, J. An, C. Yuen, and L. Gan, "Intra-class universal adversarial attacks on deep learning-based modulation classifiers," *IEEE Commun. Lett.*, vol. 27, no. 5, pp. 1297–1301, 2023.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, Long Beach, CA, USA, Dec. 2017.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. ICLR*, Virtual Event, Austria, May. 2021.

[10] Y. Chen, B. Dong, C. Liu, W. Xiong, and S. Li, "Abandon locality: Frame-wise embedding aided transformer for automatic modulation recognition," *IEEE Commun. Lett.*, vol. 27, no. 1, pp. 327–331, Jan. 2023.

[11] J. Cai, F. Gan, X. Cao, and W. Liu, "Signal modulation classification based on the transformer network," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 8, no. 3, pp. 1348–1357, 2022.

[12] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diega, CA, USA, May. 2015.

[13] G. B. Tunze, T. Huynh-The, J.-M. Lee, and D.-S. Kim, "Sparsely connected CNN for efficient automatic modulation recognition," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15 557–15 568, Dec. 2020.

[14] J. Xu, C. Luo, G. Parr, and Y. Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1629–1632, Oct. 2020.